REMARKS

Information Disclosure Statement

The Examiner requested copies of some references mentioned in the specification. In response, the applicants enclose the requested references.

The Objection to the Drawings

The drawings were objected to because the description of FIGs. 3D and 3E on page 14, lines 18-20 calls for red curves and the drawings are in black in white. In response, the applicants have amended the specification to clarify which curves are being referred to. Acceptance of this change is courteously requested.

The Section 37 CFR 1.75 Objection of Claims 1-12 and 21-28.

Claims 1-12 and 21-28 are objected to under 37 CFR 1.75, as being indefinite. The applicants have amended Claims 1 and 11 to provide proper antecedent basis. It is believed that the foregoing amendments to Claims 1 and 11 have clarified any indefiniteness that existed in the original claim language.

It is believed the amended claims now fulfill the requirements of 37 CFR 1.75, as they particularly point out and distinctly claim the subject matter which the applicant regards as the invention. Therefore, it is respectfully requested that the objection to Claims 1-12 and 21-28 be reconsidered based on the amended claim language.

The 35 USC 102 Rejection of 1-3, 6, 7 and 12.

Claims 1-3, 6, 7 and 12 were rejected under 35 USC 102(b) as being anticipated by Wang et al., U.S. Patent No. 5,557,684 A, herein after referred to as

Wang. It was contended in the above-identified Office Action that Wang teaches all the elements of the rejected claims.

While no admission is made that these claims are actually anticipated by the cited reference, the applicants have chosen to incorporate the subject matter of Claim 8 into Claim 1 to further the prosecution of the application and expedite its allowance. Claim 8, was stated in the Office Action as being allowable if rewritten in independent form including all limitations of the base claim and any intervening claims. This is essentially what has been done by incorporating the subject matter of Claim 8 into Claim 1. In addition, Claim 9 was made dependent from Claim 1, to conform it to the foregoing change. Since the incorporation of the subject matter of Claim 8 into Claim 1 now makes Claim 1 allowable, all the rejected claims that have not been cancelled, Claims 1-7, 9-12 and 21-28 which ultimately depend from Claim 1, are allowable as well. Claim 8 has been cancelled.

The 35 USC 103 Rejection of Claims 5 and 10.

Claims 5 and 10 were rejected under 35 USC 103(a) as unpatentable over Wang et al., U.S. Patent No. 5,557,684 A), in view of Kaup et al.

While no admission is made that these claims are actually made obvious by the cited references, the applicants have chosen to incorporate the subject matter of Claim 8 into Claim 1 to further the prosecution of the application and expedite its allowance, as discussed above. Claims 5 and 10 were made dependent from Claim 1, to conform it to the foregoing change. Since the incorporation of the subject matter of Claim 8 into Claim 1 now makes Claim 1 allowable, all the rejected claims that have not been cancelled, Claims 1-7, 9-12 and 21-28 which ultimately depend from Claim 1, are allowable as well.

From *AAAI-90 Workshop on Qualitative Vision*, July 20, 1990, Boston, MA.

Ordinal characteristics of transparency.

Edward H. Adelson* and P. Anandan**

*Media Laboratory and Department of Brain and Cognitive Sciences, MIT, Cambridge MA 02139, and **Department of Computer Science, Yale University, New Haven CT 06520

Figure 1 shows an example of visual transparency. The image could arise from a number of different physical causes. For example, a square of tissue paper could be in front of a dark grey circle; or a circular shadow could be cast on a plane containing a light grey square; or a dark circular filter could be lying on top of a light grey square. Although the physics is uncertain, one can perceive the image as a combination of two more primitive images.

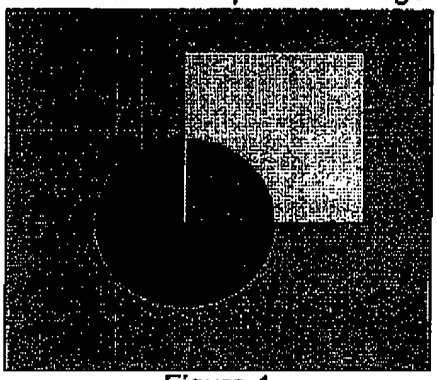


Figure 1

We use the term "transparency" to cover the general case of such image combination, including what would be called "translucency" in ordinary language. Many physical phenomena can produce transparency. For example, dark filters, specular reflections, puffs of smoke, gauze curtains, and cast shadows, all combine with patterns behind them in a transparent manner.

When an image has been formed by the combination of two primitive images, then it is usually more parsimonious to describe the image in terms of that combination; thus it is advantageous for a visual system to parse the image into the primitive images along with a combination rule. This parsimony does not depend on assigning a unique physical interpretation to the primitive images; figure 1 can be parsed into a circle and a square, even in the absence of a decision about the underlying physics.

We suggest that visual transparency may be initially analysed at a "pre-physical" level, which does not include the physical specificity of a full intrinsic image analysis [1]. The representation at this level consists of a set of primitive image layers which are ordered in depth. Each layer contains filled regions which modify the appearance of the layers beneath them, and unfilled regions which are perfectly clear. The filled regions of different layers combine with

each other according to simple rules such as multiplication and addition. Figure 2 shows an example of the layers which might give rise to figure 1. In the simplest layered model luminance only propogates from the lower layers to the higher ones (i.e. toward the viewer).

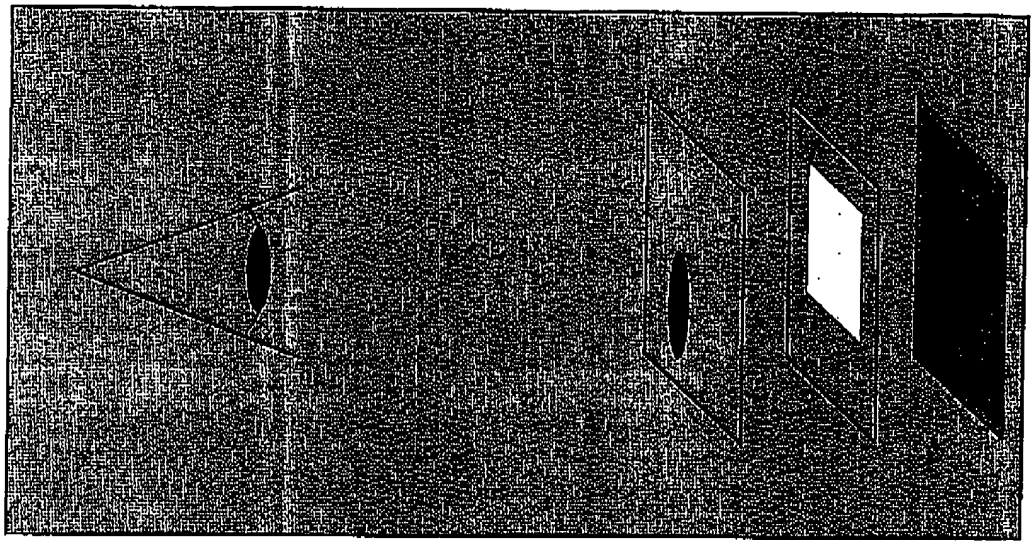


Figure 2

The significance of X junctions.

When patterns on distinct layers overlap, they typically give rise to X junctions in the image, which have an important influence on the perception of transparency by humans [2 - 4]. These X junctions can be quite diagnostic of the nature of the transparent interaction, and the depth ordering of layers. For example, figures 3(a-c) contain three images, which the human visual system interprets in three different ways. Transparency can be seen in figure 3(a), which is interpreted as containing two dark filters; however, the depth ordering is ambiguous: either square can be seen as lying in front of the other. Transparency can also be seen in figure 3(b), but in this case the depth ordering is unambiguous: the square on the lower-left appears to be in front. Finally, transparency is not seen in figure 3(c), which is commonly seen as a painted pattern lying in a single layer.

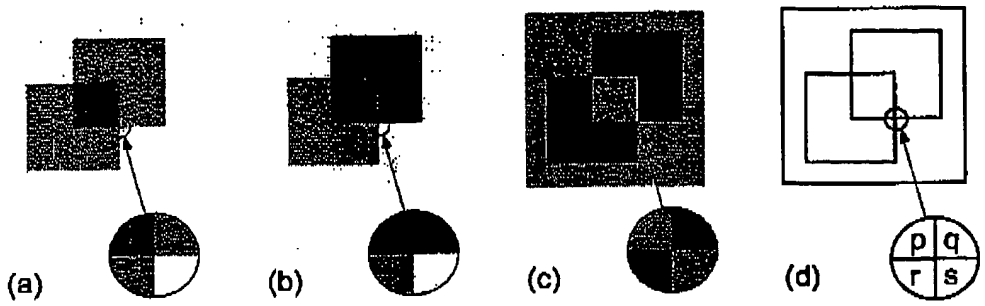


Figure 3

The qualitative characteristics of the transparency can be related to the qualitative characteristics of the X junctions. Let p,q,r,s be the luminances in the four regions surrounding the X junction, as indicated in figure 3(d). In figure 3(a) $p<q$ and $r<s$, which is to say that the vertical edge retains the same sign in both halves of the X junction. Similarly, $p<r$ and $q<s$, which is to say that horizontal edge, also retains the same sign in both halves of the X junction. We call this a "non-reversing" junction because both edges retain their sign. In figure 3(b) the vertical edge changes sign across the X junction, whereas the horizontal edge retains its sign. We call this a "single-reversing" junction. Finally, in figure 3(c) both the horizontal and the vertical edges change sign across the X junction. We call this a "double-reversing" junction.

The human visual system seems to employ heuristics related to these different categories of X junctions. Non-reversing junctions support the perception of transparency, while leaving the depth ordering of the layers ambiguous. Single-reversing junctions also support transparency, and in addition impose a unique depth ordering. Double-reversing junctions do not support transparency. The junctions thus offer pieces of local evidence which may be propogated through the figure to the interpretation of transparency and depth order.

Computational analysis

We may examine transparency from a computational point of view, to understand the basis for the heuristics described above. We begin with a framework to characterize the combination of transparent layers; the layers will be denoted $l_1 \ldots l_n$. Each layer may attenuate the luminance from the layer beneath it by a factor a, $0 < a \leq 1$, and may contribute its own emission of quantity e, $e \geq 0$. The attenuation and emission are functions of position, $a(x,y)$ and $e(x,y)$. An unfilled region has $a = 1$ and $e = 0$. (This formulation is slightly different from Metelli's, but the resulting restrictions are similar to those derived from Metelli's rules).

If layer n-1 contains a luminance pattern $l_{n-1}(x,y)$, then the luminance pattern at layer n is:

$$l_n(x,y) = a_n(x,y).l_{n-1}(x,y) + e_n(x,y) \qquad (1)$$

Since both multiplicative and additive interactions are allowed, a wide range of (p,q,r,s) values are legal examples of transparency. However, there are constraints on those values. The allowed ranges of a and e imply that a filled region must reduce or leave unchanged the amplitude of the luminance variation in a lower layer. This allows us to establish some inequalities concerning the X junction of figure 3(d). We assume that an X-junction results from the overlap of filled regions in two layers; it remains to determine whether the frontal layer's edge is vertical or horizontal, and which half of the edge is filled.

The four possible local hypotheses about the filled frontal region are: (i) it lies above the horizontal line, (ii) it lies below the horizontal line, (iii) it lies to the left of the vertical line, and (iv) it lies to the right of the vertical line. The conditions on the attenuation factor translate into the following inequality conditions,

(1) hypothesis (i) is physically plausible iff $0< (p-q)/(r-s) \leq 1$,
(2) hypothesis (ii) is plausible iff $0< (r-s)/(p-q) \leq 1$,
(3) hypothesis (iii) is plausible iff $0 < (p-r)/(q-s) \leq 1$, and
(4) hypothesis (iv) is plausible iff $0 < (q-s)/(p-r) \leq 1$.

Note that conditions (i) and (ii) are mutually exclusive unless the ratio is unity, likewise for conditions (iii) and (iv).

The fact that these ratios are non-negative leads to the edge-reversal heuristics noted above. Thus an edge which is tranparently occluded cannot reverse sign, while an edge which is in front may or may not reverse sign. It follows that double-reversing junctions have two consistent interpretations, and single-reversing junctions have only one. A double-reversing junction would require that both the vertical and horizontal edges be in front of the other, which is impossible; therefore no transparent interpretation is allowed.

Conclusion

Transparency can arise in images due to a number of different physical phenomena. We have proposed a pre-physical level of representation in which a number primitive images organized as layers combine together to form an observed image. The ordinal relationships between the luminances at an X junction can be used categorize the X junction as non-,single-,and double-reversing junctions. These categories can be determined without precise measurements, and are robust against point nonlinearities in luminance sensitivity. Non- and single-reversing junctions support transparency; single-reversing junctions lead to an unambiguous interpretation of depth-order of the layers, while non-reversing junctions leave the depth-order ambiguous. Double-reversing junctions do not support transparency. Propogation of these constraints can be used to rapidly restrict the set of the legal interpretations of an image.

References:

[1] Barrow, H. G., and Tenenbaum, J. M., "Recovering Intrinsic Scene Characteristics from Images." In Hanson, A. R., and Riseman, E. M., **Computer Vision Systems**, 3-26 (1978).

[2] Beck, J., Prazdny, K., and Ivry, R., "The perception of transparency with achromatic colors." *Perception and Psychophysics*, **35**, 407-422 (1984).

[3] Kersten, D., "Transparency and the Cooperative Computation of Scene Attributes", to appear in Landy, M., and Movshon, J., eds., **Computational Models of Visual Processing**, MIT Press, 1990.

[4] Metelli, F., "The perception of transparency," *Scientific American*, **230**, (4), 91-98

Recovering High Dynamic Range Radiance Maps from Photographs

Paul E. Debevec Jitendra Malik

University of California at Berkeley[1]

ABSTRACT

We present a method of recovering high dynamic range radiance maps from photographs taken with conventional imaging equipment. In our method, multiple photographs of the scene are taken with different amounts of exposure. Our algorithm uses these differently exposed photographs to recover the response function of the imaging process, up to factor of scale, using the assumption of reciprocity. With the known response function, the algorithm can fuse the multiple photographs into a single, high dynamic range radiance map whose pixel values are proportional to the true radiance values in the scene. We demonstrate our method on images acquired with both photochemical and digital imaging processes. We discuss how this work is applicable in many areas of computer graphics involving digitized photographs, including image-based modeling, image compositing, and image processing. Lastly, we demonstrate a few applications of having high dynamic range radiance maps, such as synthesizing realistic motion blur and simulating the response of the human visual system.

CR Descriptors: I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding - *Intensity, color, photometry and thresholding*; I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism - *Color, shading, shadowing, and texture*; I.4.1 [**Image Processing**]: Digitization - *Scanning*; I.4.8 [**Image Processing**]: Scene Analysis - *Photometry, Sensor Fusion.*

1 Introduction

Digitized photographs are becoming increasingly important in computer graphics. More than ever, scanned images are used as texture maps for geometric models, and recent work in image-based modeling and rendering uses images as the fundamental modeling primitive. Furthermore, many of today's graphics applications require computer-generated images to mesh seamlessly with real photographic imagery. Properly using photographically acquired imagery in these applications can greatly benefit from an accurate model of the photographic process.

When we photograph a scene, either with film or an electronic imaging array, and digitize the photograph to obtain a two-dimensional array of "brightness" values, these values are rarely true measurements of relative radiance in the scene. For example, if one pixel has twice the value of another, it is unlikely that it observed twice the radiance. Instead, there is usually an unknown, nonlinear mapping that determines how radiance in the scene becomes pixel values in the image.

This nonlinear mapping is hard to know beforehand because it is actually the composition of several nonlinear mappings that occur in the photographic process. In a conventional camera (see Fig. 1), the film is first exposed to light to form a latent image. The film is then developed to change this latent image into variations in transparency, or *density*, on the film. The film can then be digitized using a film scanner, which projects light through the film onto an electronic light-sensitive array, converting the image to electrical voltages. These voltages are digitized, and then manipulated before finally being written to the storage medium. If prints of the film are scanned rather than the film itself, then the printing process can also introduce nonlinear mappings.

In the first stage of the process, the film response to variations in exposure X (which is $E\Delta t$, the product of the irradiance E the film receives and the exposure time Δt) is a non-linear function, called the "characteristic curve" of the film. Noteworthy in the typical characteristic curve is the presence of a small response with no exposure and saturation at high exposures. The development, scanning and digitization processes usually introduce their own nonlinearities which compose to give the aggregate nonlinear relationship between the image pixel exposures X and their values Z.

Digital cameras, which use charge coupled device (CCD) arrays to image the scene, are prone to the same difficulties. Although the charge collected by a CCD element is proportional to its irradiance, most digital cameras apply a nonlinear mapping to the CCD outputs before they are written to the storage medium. This nonlinear mapping is used in various ways to mimic the response characteristics of film, anticipate nonlinear responses in the display device, and often to convert 12-bit output from the CCD's analog-to-digital converters to 8-bit values commonly used to store images. As with film, the most significant nonlinearity in the response curve is at its saturation point, where any pixel with a radiance above a certain level is mapped to the same maximum image value.

Why is this any problem at all? The most obvious difficulty, as any amateur or professional photographer knows, is that of limited dynamic range—one has to choose the range of radiance values that are of interest and determine the exposure time suitably. Sunlit scenes, and scenes with shiny materials and artificial light sources, often have extreme differences in radiance values that are impossible to capture without either under-exposing or saturating the film. To cover the full dynamic range in such a scene, one can take a series of photographs with different exposures. This then poses a problem: how can we combine these separate images into a composite radiance map? Here the fact that the mapping from scene radiance to pixel values is unknown and nonlinear begins to haunt us. The purpose of this paper is to present a simple technique for recovering this response function, up to a scale factor, using nothing more than a set of photographs taken with varying, known exposure durations. With this mapping, we then use the pixel values from all available photographs to construct an accurate map of the radiance in the scene, up to a factor of scale. This radiance map will cover

[1]Computer Science Division, University of California at Berkeley, Berkeley, CA 94720-1776. Email: debevec@cs.berkeley.edu, malik@cs.berkeley.edu. More information and additional results may be found at: http://www.cs.berkeley.edu/~debevec/Research

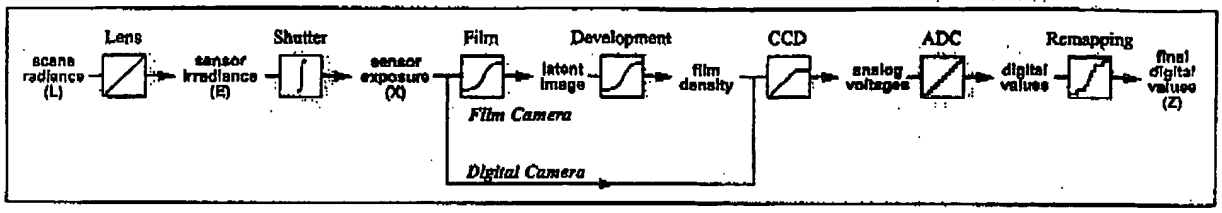


Figure 1: **Image Acquisition Pipeline** *shows how scene radiance becomes pixel values for both film and digital cameras. Unknown nonlinear mappings can occur during exposure, development, scanning, digitization, and remapping. The algorithm in this paper determines the aggregate mapping from scene radiance L to pixel values Z from a set of differently exposed images.*

the entire dynamic range captured by the original photographs.

1.1 Applications

Our technique of deriving imaging response functions and recovering high dynamic range radiance maps has many possible applications in computer graphics:

Image-based modeling and rendering

Image-based modeling and rendering systems to date (e.g. [11, 15, 2, 3, 12, 6, 17]) make the assumption that all the images are taken with the same exposure settings and film response functions. However, almost any large-scale environment will have some areas that are much brighter than others, making it impossible to adequately photograph the scene using a single exposure setting. In indoor scenes with windows, this situation often arises within the field of view of a single photograph, since the areas visible through the windows can be far brighter than the areas inside the building.

By determining the response functions of the imaging device, the method presented here allows one to correctly fuse pixel data from photographs taken at different exposure settings. As a result, one can properly photograph outdoor areas with short exposures, and indoor areas with longer exposures, without creating inconsistencies in the data set. Furthermore, knowing the response functions can be helpful in merging photographs taken with different imaging systems, such as video cameras, digital cameras, and film cameras with various film stocks and digitization processes.

The area of image-based modeling and rendering is working toward recovering more advanced reflection models (up to complete BRDF's) of the surfaces in the scene (e.g. [21]). These methods, which involve observing surface radiance in various directions under various lighting conditions, require absolute radiance values rather than the nonlinearly mapped pixel values found in conventional images. Just as important, the recovery of high dynamic range images will allow these methods to obtain accurate radiance values from surface specularities and from incident light sources. Such higher radiance values usually become clamped in conventional images.

Image processing

Most image processing operations, such as blurring, edge detection, color correction, and image correspondence, expect pixel values to be proportional to the scene radiance. Because of nonlinear image response, especially at the point of saturation, these operations can produce incorrect results for conventional images.

In computer graphics, one common image processing operation is the application of synthetic motion blur to images. In our results (Section 3), we will show that using true radiance maps produces significantly more realistic motion blur effects for high dynamic range scenes.

Image compositing

Many applications in computer graphics involve compositing image data from images obtained by different processes. For example, a background matte might be shot with a still camera, live action might be shot with a different film stock or scanning process, and CG elements would be produced by rendering algorithms. When there are significant differences in the response curves of these imaging processes, the composite image can be visually unconvincing. The technique presented in this paper provides a convenient and robust method of determining the overall response curve of any imaging process, allowing images from different processes to be used consistently as radiance maps. Furthermore, the recovered response curves can be inverted to render the composite radiance map as if it had been photographed with any of the original imaging processes, or a different imaging process entirely.

A research tool

One goal of computer graphics is to simulate the image formation process in a way that produces results that are consistent with what happens in the real world. Recovering radiance maps of real-world scenes should allow more quantitative evaluations of rendering algorithms to be made in addition to the qualitative scrutiny they traditionally receive. In particular, the method should be useful for developing reflectance and illumination models, and comparing global illumination solutions against ground truth data.

Rendering high dynamic range scenes on conventional display devices is the subject of considerable previous work, including [20, 16, 5, 23]. The work presented in this paper will allow such methods to be tested on real radiance maps in addition to synthetically computed radiance solutions.

1.2 Background

The photochemical processes involved in silver halide photography have been the subject of continued innovation and research ever since the invention of the daguerretype in 1839. [18] and [8] provide a comprehensive treatment of the theory and mechanisms involved. For the newer technology of solid-state imaging with charge coupled devices, [19] is an excellent reference. The technical and artistic problem of representing the dynamic range of a natural scene on the limited range of film has concerned photographers from the early days – [1] presents one of the best known systems to choose shutter speeds, lens apertures, and developing conditions to best coerce the dynamic range of a scene to fit into what is possible on a print. In scientific applications of photography, such as in astronomy, the nonlinear film response has been addressed by suitable calibration procedures. It is our objective instead to develop a simple self-calibrating procedure not requiring calibration charts or photometric measuring devices.

In previous work, [13] used multiple flux integration times of a CCD array to acquire extended dynamic range images. Since direct CCD outputs were available, the work did not need to deal with the

problem of nonlinear pixel value response. [14] addressed the problem of nonlinear response but provide a rather limited method of recovering the response curve. Specifically, a parametric form of the response curve is arbitrarily assumed, there is no satisfactory treatment of image noise, and the recovery process makes only partial use of the available data.

2 The Algorithm

This section presents our algorithm for recovering the film response function, and then presents our method of reconstructing the high dynamic range radiance image from the multiple photographs. We describe the algorithm assuming a grayscale imaging device. We discuss how to deal with color in Section 2.6.

2.1 Film Response Recovery

Our algorithm is based on exploiting a physical property of imaging systems, both photochemical and electronic, known as *reciprocity*.

Let us consider photographic film first. The response of a film to variations in exposure is summarized by the characteristic curve (or Hurter-Driffield curve). This is a graph of the optical density D of the processed film against the logarithm of the exposure X to which it has been subjected. The exposure X is defined as the product of the irradiance E at the film and exposure time, Δt, so that its units are Jm^{-2}. Key to the very concept of the characteristic curve is the assumption that only the product $E\Delta t$ is important, that halving E and doubling Δt will not change the resulting optical density D. Under extreme conditions (very large or very low Δt), the reciprocity assumption can break down, a situation described as reciprocity failure. In typical print films, reciprocity holds to within $\frac{1}{3}$ stop[1] for exposure times of 10 seconds to 1/10,000 of a second.[2] In the case of charge coupled arrays, reciprocity holds under the assumption that each site measures the total number of photons it absorbs during the integration time.

After the development, scanning and digitization processes, we obtain a digital number Z, which is a nonlinear function of the original exposure X at the pixel. Let us call this function f, which is the composition of the characteristic curve of the film as well as all the nonlinearities introduced by the later processing steps. Our first goal will be to recover this function f. Once we have that, we can compute the exposure X at each pixel, as $X = f^{-1}(Z)$. We make the reasonable assumption that the function f is monotonically increasing, so its inverse f^{-1} is well defined. Knowing the exposure X and the exposure time Δt, the irradiance E is recovered as $E = X/\Delta t$, which we will take to be proportional to the radiance L in the scene.[3]

Before proceeding further, we should discuss the consequences of the spectral response of the sensor. The exposure X should be thought of as a function of wavelength $X(\lambda)$, and the abscissa on the characteristic curve should be the integral $\int X(\lambda)R(\lambda)d\lambda$ where $R(\lambda)$ is the spectral response of the sensing element at the pixel location. Strictly speaking, our use of irradiance, a radiometric quantity, is not justified. However, the spectral response of the sensor site may not be the photopic luminosity function V_λ, so the photometric term *illumtnance* is not justified either. In what follows, we will use the term irradiance, while urging the reader to remember that the quantities we will be dealing with are weighted by the spectral response at the sensor site. For color photography, the color channels may be treated separately.

The input to our algorithm is a number of digitized photographs taken from the same vantage point with different known exposure durations Δt_j.[4] We will assume that the scene is static and that this process is completed quickly enough that lighting changes can be safely ignored. It can then be assumed that the film irradiance values E_i for each pixel i are constant. We will denote pixel values by Z_{ij} where i is a spatial index over pixels and j indexes over exposure times Δt_j. We may now write down the film reciprocity equation as:

$$Z_{ij} = f(E_i\Delta t_j) \tag{1}$$

Since we assume f is monotonic, it is invertible, and we can rewrite (1) as:

$$f^{-1}(Z_{ij}) = E_i\Delta t_j$$

Taking the natural logarithm of both sides, we have:

$$\ln f^{-1}(Z_{ij}) = \ln E_i + \ln \Delta t_j$$

To simplify notation, let us define function $g = \ln f^{-1}$. We then have the set of equations:

$$g(Z_{ij}) = \ln E_i + \ln \Delta t_j \tag{2}$$

where i ranges over pixels and j ranges over exposure durations. In this set of equations, the Z_{ij} are known, as are the Δt_j. The unknowns are the irradiances E_i, as well as the function g, although we assume that g is smooth and monotonic.

We wish to recover the function g and the irradiances E_i that best satisfy the set of equations arising from Equation 2 in a least-squared error sense. We note that recovering g only requires recovering the *finite* number of values that $g(z)$ can take since the domain of Z, pixel brightness values, is finite. Letting Z_{min} and Z_{max} be the least and greatest pixel values (integers), N be the number of pixel locations and P be the number of photographs, we formulate the problem as one of finding the $(Z_{max} - Z_{min} + 1)$ values of $g(Z)$ and the N values of $\ln E_i$ that minimize the following quadratic objective function:

$$\mathcal{O} = \sum_{i=1}^{N}\sum_{j=1}^{P}[g(Z_{ij}) - \ln E_i - \ln \Delta t_j]^2 + \lambda \sum_{z=Z_{min}+1}^{Z_{max}-1} g''(z)^2 \tag{3}$$

The first term ensures that the solution satisfies the set of equations arising from Equation 2 in a least squares sense. The second term is a smoothness term on the sum of squared values of the second derivative of g to ensure that the function g is smooth; in this discrete setting we use $g''(z) = g(z-1) - 2g(z) + g(z+1)$. This smoothness term is essential to the formulation in that it provides coupling between the values $g(z)$ in the minimization. The scalar λ weights the smoothness term relative to the data fitting term, and should be chosen appropriately for the amount of noise expected in the Z_{ij} measurements.

Because it is quadratic in the E_i's and $g(z)$'s, minimizing $\mathcal{O}$ is a straightforward linear least squares problem. The overdetermined

[1] 1 stop is a photographic term for a factor of two; $\frac{1}{3}$ stop is thus $2^{\frac{1}{3}}$

[2] An even larger dynamic range can be covered by using neutral density filters to lessen to amount of light reaching the film for a given exposure time. A discussion of the modes of reciprocity failure may be found in [18], ch. 4.

[3] L is proportional E for any particular pixel, but it is possible for the proportionality factor to be different at different places on the sensor. One formula for this variance, given in [7], is $E = L\frac{\pi}{4}\left(\frac{d}{f}\right)^2 \cos^4\alpha$, where α measures the pixel's angle from the lens' optical axis. However, most modern camera lenses are designed to compensate for this effect, and provide a nearly constant mapping between radiance and irradiance at f/8 and smaller apertures. See also [10].

[4] Most modern SLR cameras have electronically controlled shutters which give extremely accurate and reproducible exposure times. We tested our Canon EOS Elan camera by using a Macintosh to make digital audio recordings of the shutter. By analyzing these recordings we were able to verify the accuracy of the exposure times to within a thousandth of a second. Conveniently, we determined that the actual exposure times varied by powers of two between stops ($\frac{1}{64}$, $\frac{1}{32}$, $\frac{1}{16}$, $\frac{1}{8}$, $\frac{1}{4}$, $\frac{1}{2}$, 1, 2, 4, 8, 16, 32), rather than the rounded numbers displayed on the camera readout ($\frac{1}{60}$, $\frac{1}{30}$, $\frac{1}{15}$, $\frac{1}{8}$, $\frac{1}{4}$, $\frac{1}{2}$, 1, 2, 4, 8, 15, 30). Because of problems associated with vignetting, varying the aperture is not recommended.

system of linear equations is robustly solved using the singular value decomposition (SVD) method. An intuitive explanation of the procedure may be found in Fig. 2.

We need to make three additional points to complete our description of the algorithm:

First, the solution for the $g(z)$ and E_i values can only be up to a single scale factor α. If each log irradiance value $\ln E_i$ were replaced by $\ln E_i + \alpha$, and the function g replaced by $g + \alpha$, the system of equations 2 and also the objective function $\mathcal{O}$ would remain unchanged. To establish a scale factor, we introduce the additional constraint $g(Z_{mid}) = 0$, where $Z_{mid} = \frac{1}{2}(Z_{min} + Z_{max})$, simply by adding this as an equation in the linear system. The meaning of this constraint is that a pixel with value midway between Z_{min} and Z_{max} will be assumed to have unit exposure.

Second, the solution can be made to have a much better fit by anticipating the basic shape of the response function. Since $g(z)$ will typically have a steep slope near Z_{min} and Z_{max}, we should expect that $g(z)$ will be less smooth and will fit the data more poorly near these extremes. To recognize this, we can introduce a weighting function $w(z)$ to emphasize the smoothness and fitting terms toward the middle of the curve. A sensible choice of w is a simple hat function:

$$w(z) = \begin{cases} z - Z_{min} & \text{for } z \leq \frac{1}{2}(Z_{min} + Z_{max}) \\ Z_{max} - z & \text{for } z > \frac{1}{2}(Z_{min} + Z_{max}) \end{cases} \quad (4)$$

Equation 3 now becomes:

$$\mathcal{O} = \sum_{i=1}^{N} \sum_{j=1}^{P} \{w(Z_{ij})[g(Z_{ij}) - \ln E_i - \ln \Delta t_j]\}^2 + \lambda \sum_{z=Z_{min}+1}^{Z_{max}-1} [w(z)g''(z)]^2$$

Finally, we need not use every available pixel site in this solution procedure. Given measurements of N pixels in P photographs, we have to solve for N values of $\ln E_i$ and $(Z_{max} - Z_{min})$ samples of g. To ensure a sufficiently overdetermined system, we want $N(P-1) > (Z_{max} - Z_{min})$. For the pixel value range $(Z_{max} - Z_{min}) = 255$, $P = 11$ photographs, a choice of N on the order of 50 pixels is more than adequate. Since the size of the system of linear equations arising from Equation 3 is on the order of $N \times P + Z_{max} - Z_{min}$, computational complexity considerations make it impractical to use every pixel location in this algorithm. Clearly, the pixel locations should be chosen so that they have a reasonably even distribution of pixel values from Z_{min} to Z_{max}, and so that they are spatially well distributed in the image. Furthermore, the pixels are best sampled from regions of the image with low intensity variance so that radiance can be assumed to be constant across the area of the pixel, and the effect of optical blur of the imaging system is minimized. So far we have performed this task by hand, though it could easily be automated.

Note that we have not explicitly enforced the constraint that g must be a monotonic function. If desired, this can be done by transforming the problem to a non-negative least squares problem. We have not found it necessary because, in our experience, the smoothness penalty term is enough to make the estimated g monotonic in addition to being smooth.

To show its simplicity, the MATLAB routine we used to minimize Equation 5 is included in the Appendix. Running times are on the order of a few seconds.

2.2 Constructing the High Dynamic Range Radiance Map

Once the response curve g is recovered, it can be used to quickly convert pixel values to relative radiance values, assuming the exposure Δt_j is known. Note that the curve can be used to determine radiance values in any image(s) acquired by the imaging process associated with g, not just the images used to recover the response function.

From Equation 2, we obtain:

$$\ln E_i = g(Z_{ij}) - \ln \Delta t_j \quad (5)$$

For robustness, and to recover high dynamic range radiance values, we should use all the available exposures for a particular pixel to compute its radiance. For this, we reuse the weighting function in Equation 4 to give higher weight to exposures in which the pixel's value is closer to the middle of the response function:

$$\ln E_i = \frac{\sum_{j=1}^{P} w(Z_{ij})(g(Z_{ij}) - \ln \Delta t_j)}{\sum_{j=1}^{P} w(Z_{ij})} \quad (6)$$

Combining the multiple exposures has the effect of reducing noise in the recovered radiance values. It also reduces the effects of imaging artifacts such as film grain. Since the weighting function ignores saturated pixel values, "blooming" artifacts[5] have little impact on the reconstructed radiance values.

2.2.1 Storage

In our implementation the recovered radiance map is computed as an array of single-precision floating point values. For efficiency, the map can be converted to the image format used in the RADIANCE [22] simulation and rendering system, which uses just eight bits for each of the mantissa and exponent. This format is particularly compact for color radiance maps, since it stores just one exponent value for all three color values at each pixel. Thus, in this format, a high dynamic range radiance map requires just one third more storage than a conventional RGB image.

2.3 How many images are necessary?

To decide on the number of images needed for the technique, it is convenient to consider the two aspects of the process:

1. *Recovering the film response curve:* This requires a minimum of two photographs. Whether two photographs are enough can be understood in terms of the heuristic explanation of the process of film response curve recovery shown in Fig. 2. If the scene has sufficiently many different radiance values, the entire curve can, in principle, be assembled by sliding together the sampled curve segments, each with only two samples. Note that the photos must be similar enough in their exposure amounts that some pixels fall into the working range[6] of the film in both images; otherwise, there is no information to relate the exposures to each other. Obviously, using more than two images with differing exposure times improves performance with respect to noise sensitivity.
2. *Recovering a radiance map given the film response curve:* The number of photographs needed here is a function of the dynamic range of radiance values in the scene. Suppose the range of maximum to minimum radiance values that we are

[5] Blooming occurs when charge or light at highly saturated sites on the imaging surface spills over and affects values at neighboring sites.

[6] The *working range* of the film corresponds to the middle section of the response curve. The ends of the curve, in which large changes in exposure cause only small changes in density (or pixel value), are called the *toe* and the *shoulder*.

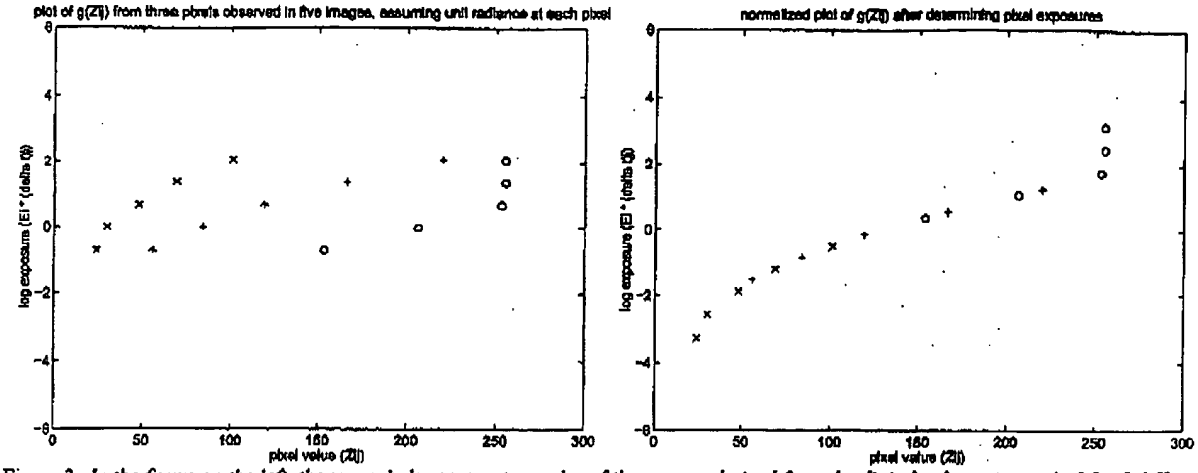


Figure 2: *In the figure on the left, the × symbols represent samples of the g curve derived from the digital values at one pixel for 5 different known exposures using Equation 2. The unknown log irradiance* $\ln E_i$ *has been arbitrarily assumed to be 0. Note that the shape of the g curve is correct, though its position on the vertical scale is arbitrary corresponding to the unknown* $\ln E_i$*. The + and ○ symbols show samples of g curve segments derived by consideration of two other pixels; again the vertical position of each segment is arbitrary. Essentially, what we want to achieve in the optimization process is to slide the 3 sampled curve segments up and down (by adjusting their* $\ln E_i$*'s) until they "line up" into a single smooth, monotonic curve, as shown in the right figure. The vertical position of the composite curve will remain arbitrary.*

interested in recovering accurately is R, and the film is capable of representing in its working range a dynamic range of F. Then the minimum number of photographs needed is $\lceil \frac{R}{F} \rceil$ to ensure that every part of the scene is imaged in at least one photograph at an exposure duration that puts it in the working range of the film response curve. As in recovering the response curve, using more photographs than strictly necessary will result in better noise sensitivity.

If one wanted to use as few photographs as possible, one might first recover the response curve of the imaging process by photographing a scene containing a diverse range of radiance values at three or four different exposures, differing by perhaps one or two stops. This response curve could be used to determine the working range of the imaging process, which for the processes we have seen would be as many as five or six stops. For the remainder of the shoot, the photographer could decide for any particular scene the number of shots necessary to cover its entire dynamic range. For diffuse indoor scenes, only one exposure might be necessary; for scenes with high dynamic range, several would be necessary. By recording the exposure amount for each shot, the images could then be converted to radiance maps using the pre-computed response curve.

2.4 Recovering extended dynamic range from single exposures

Most commericially available film scanners can detect reasonably close to the full range of useful densities present in film. However, many of these scanners (as well as the Kodak PhotoCD process) produce 8-bit-per-channel images designed to be viewed on a screen or printed on paper. Print film, however, records a significantly greater dynamic range than can be displayed with either of these media. As a result, such scanners deliver only a portion of the detected dynamic range of print film in a single scan, discarding information in either high or low density regions. The portion of the detected dynamic range that is delivered can usually be influenced by "brightness" or "density adjustment" controls.

The method presented in this paper enables two methods for recovering the full dynamic range of print film which we will briefly outline[7]. In the first method, the print negative is scanned with the scanner set to scan slide film. Most scanners will then record the entire detectable dynamic range of the film in the resulting image. As before, a series of differently exposed images of the same scene can be used to recover the response function of the imaging system with each of these scanner settings. This response function can then be used to convert individual exposures to radiance maps. Unfortunately, since the resulting image is still 8-bits-per-channel, this results in increased quantization.

In the second method, the film can be scanned twice with the scanner set to different density adjustment settings. A series of differently exposed images of the same scene can then be used to recover the response function of the imaging system at each of these density adjustment settings. These two response functions can then be used to combine two scans of any single negative using a similar technique as in Section 2.2.

2.5 Obtaining Absolute Radiance

For many applications, such as image processing and image compositing, the relative radiance values computed by our method are all that are necessary. If needed, an approximation to the scaling term necessary to convert to absolute radiance can be derived using the ASA of the film[8] and the shutter speeds and exposure amounts in the photographs. With these numbers, formulas that give an approximate prediction of film response can be found in [9]. Such an approximation can be adequate for simulating visual artifacts such as glare, and predicting areas of scotopic retinal response. If desired, one could recover the scaling factor precisely by photographing a calibration luminaire of known radiance, and scaling the radiance values to agree with the known radiance of the luminaire.

2.6 Color

Color images, consisting of red, green, and blue channels, can be processed by reconstructing the imaging system response curve for

[7] This work was done in collaboration with Gregory Ward Larson

[8] Conveniently, most digital cameras also specify their sensitivity in terms of ASA.

each channel independently. Unfortunately, there will be three unknown scaling factors relating relative radiance to absolute radiance, one for each channel. As a result, different choices of these scaling factors will change the color balance of the radiance map.

By default, the algorithm chooses the scaling factor such that a pixel with value Z_{mid} will have unit exposure. Thus, any pixel with the RGB value $(Z_{mid}, Z_{mid}, Z_{mid})$ will have equal radiance values for R, G, and B, meaning that the pixel is achromatic. If the three channels of the imaging system actually do respond equally to achromatic light in the neighborhood of Z_{mid}, then our procedure correctly reconstructs the relative radiances.

However, films are usually calibrated to respond achromatically to a particular color of light C, such as sunlight or fluorescent light. In this case, the radiance values of the three channels should be scaled so that the pixel value $(Z_{mid}, Z_{mid}, Z_{mid})$ maps to a radiance with the same color ratios as C. To properly model the color response of the entire imaging process rather than just the film response, the scaling terms can be adjusted by photographing a calibration luminaire of known color.

2.7 Taking virtual photographs

The recovered response functions can also be used to map radiance values back to pixel values for a given exposure Δt using Equation 1. This process can be thought of as taking a virtual photograph of the radiance map, in that the resulting image will exhibit the response qualities of the modeled imaging system. Note that the response functions used need not be the same response functions used to construct the original radiance map, which allows photographs acquired with one imaging process to be rendered as if they were acquired with another.[5]

3 Results

Figures 3-5 show the results of using our algorithm to determine the response curve of a DCS460 digital camera. Eleven grayscale photographs filtered down to 765×509 resolution (Fig. 3) were taken at f/8 with exposure times ranging from $\frac{1}{30}$ of a second to 30 seconds, with each image receiving twice the exposure of the previous one. The film curve recovered by our algorithm from 45 pixel locations observed across the image sequence is shown in Fig. 4. Note that although CCD image arrays naturally produce linear output, from the curve it is evident that the camera nonlinearly remaps the data, presumably to mimic the response curves found in film. The underlying registered $(E_i \Delta t_j, Z_{ij})$ data are shown as light circles underneath the curve; some outliers are due to sensor artifacts (light horizontal bands across some of the darker images.)

Fig. 5 shows the reconstructed high dynamic range radiance map. To display this map, we have taken the logarithm of the radiance values and mapped the range of these values into the range of the display. In this representation, the pixels at the light regions do not saturate, and detail in the shadow regions can be made out, indicating that all of the information from the original image sequence is present in the radiance map. The large range of values present in the radiance map (over four orders of magnitude of useful dynamic range) is shown by the values at the marked pixel locations.

Figure 6 shows sixteen photographs taken inside a church with a Canon 35mm SLR camera on Fuji 100 ASA color print film. A fisheye 15mm lens set at f/8 was used, with exposure times ranging from 30 seconds to $\frac{1}{1000}$ of a second in 1-stop increments. The film was developed professionally and scanned in using a Kodak PhotoCD film scanner. The scanner was set so that it would not individually

[5] Note that here we are assuming that the spectral response functions for each channel of the two imaging processes is the same. Also, this technique does not model many significant qualities of an imaging system such as film grain, chromatic aberration, blooming, and the modulation transfer function.

Figure 3: (a) Eleven grayscale photographs of an indoor scene acquired with a Kodak DCS460 digital camera, with shutter speeds progressing in 1-stop increments from $\frac{1}{30}$ of a second to 30 seconds.

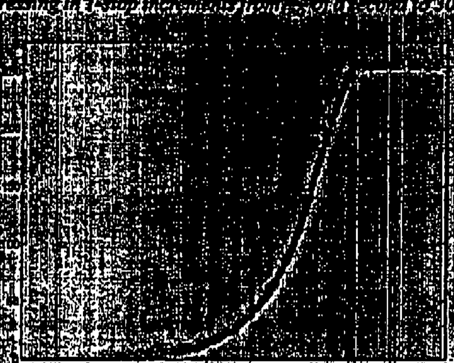


Figure 4: The response function of the DCS460 recovered by our algorithm, with the underlying $(E_i \Delta t_j, Z_{ij})$ data shown as light circles. The logarithm is base e.

Figure 5: The reconstructed high dynamic range radiance map, mapped into a grayscale image by taking the logarithm of the radiance values. The relative radiance values of the marked pixel locations, clockwise from lower left: 1.0, 46.2, 1907.1, 15116.0, and 18.0.

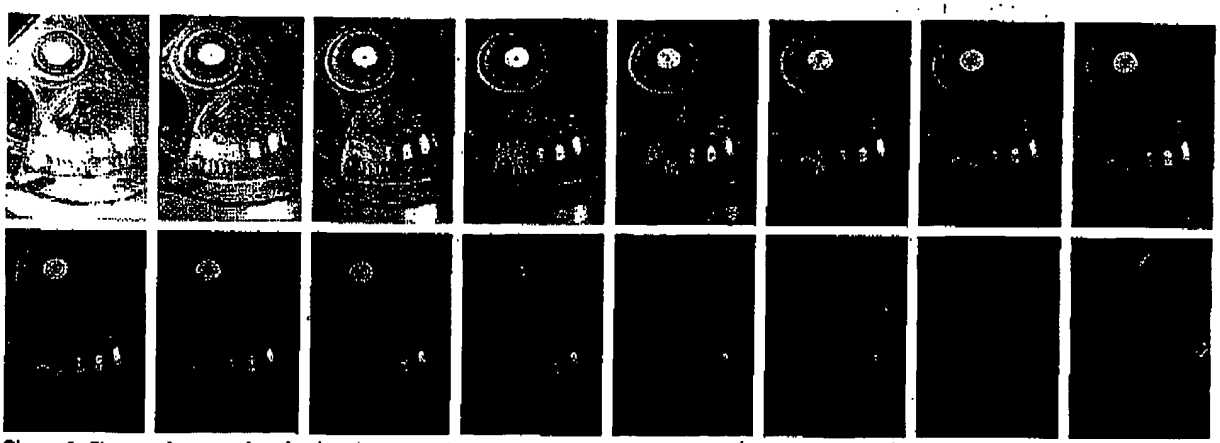


Figure 6: *Sixteen photographs of a church taken at 1-stop increments from 30 sec to $\frac{1}{1000}$ sec. The sun is directly behind the rightmost stained glass window, making it especially bright. The blue borders seen in some of the image margins are induced by the image registration process.*

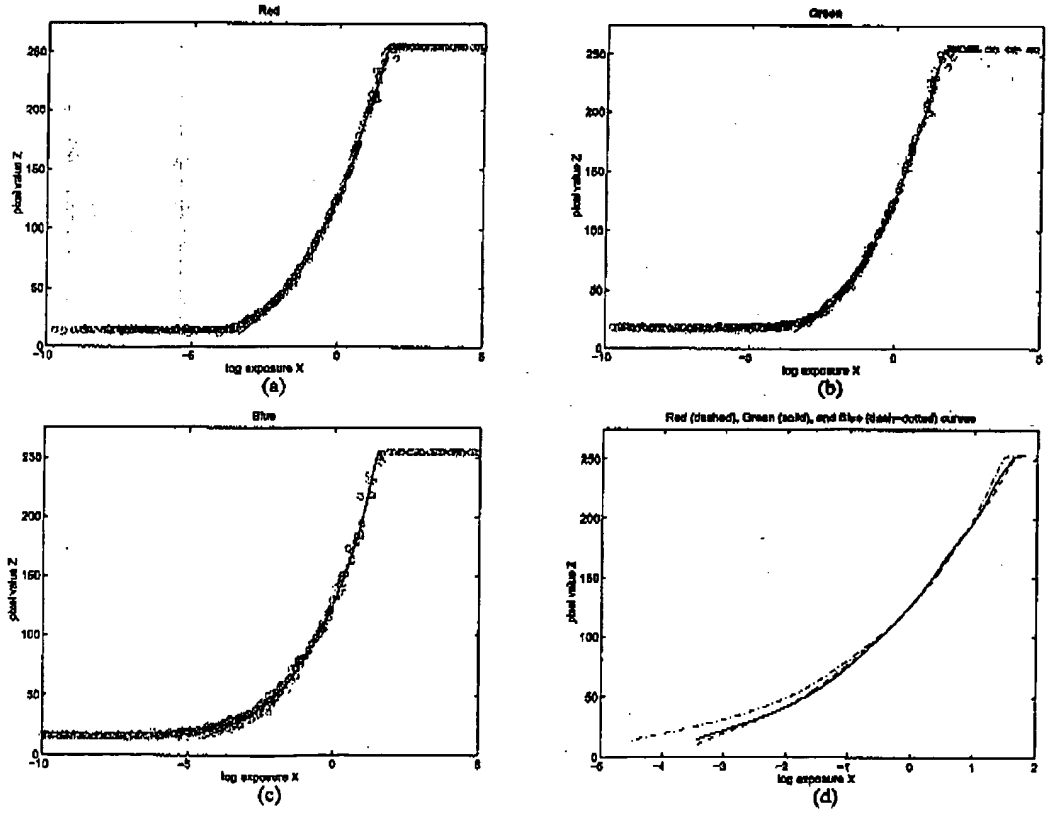
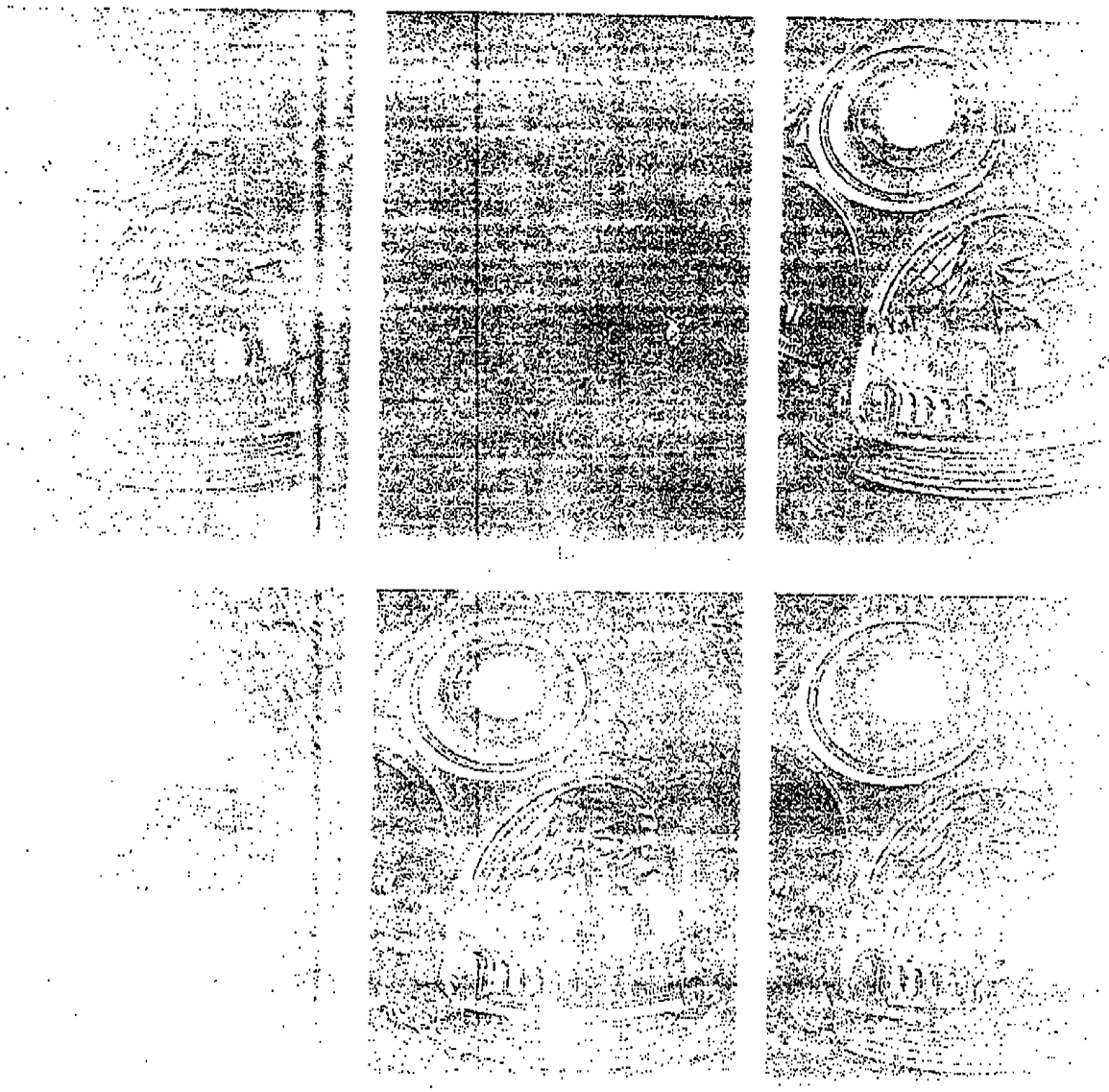


Figure 7: *Recovered response curves for the imaging system used in the church photographs in Fig. 8.* (a–c) *Response functions for the red, green, and blue channels, plotted with the underlying $(E_i \Delta t_j, Z_{ij})$ data shown as light circles.* (d) *The response functions for red, green, and blue plotted on the same axes. Note that while the red and green curves are very consistent, the blue curve rises significantly above the others for low exposure values. This indicates that dark regions in the images exhibit a slight blue cast. Since this artifact is recovered by the response curves, it does not affect the relative radiance values.*



adjust the brightness and contrast of the images[10] to guarantee that each image would be digitized using the same response function.

An unfortunate aspect of the PhotoCD process is that it does not scan precisely the same area of each negative relative to the extents of the image.[11] To counteract this effect, we geometrically registered the images to each other using a using normalized correlation (see [4]) to determine, with sub-pixel accuracy, corresponding pixels between pairs of images.

Fig. 7(a-c) shows the response functions for the red, green, and blue channels of the church sequence recovered from 28 pixel locations. Fig. 7(d) shows the recovered red, green, and blue response curves plotted on the same set of axes. From this plot, we can see that while the red and green curves are very consistent, the blue curve rises significantly above the others for low exposure values. This indicates that dark regions in the images exhibit a slight blue cast. Since this artifact is modeled by the response curves, it will not affect the relative radiance values.

Fig. 8 interprets the recovered high dynamic range radiance map in a variety of ways. Fig. 8(a) is one of the actual photographs, which lacks detail in its darker regions at the same time that many values within the two rightmost stained glass windows are saturated. Figs. 8(b,c) show the radiance map, linearly scaled to the display device using two different scaling factors. Although one scaling factor is one thousand times the other, there is useful detail in both images. Fig. 8(d) is a false-color image showing radiance values for a grayscale version of the radiance map; the highest listed radiance value is nearly 250,000 times that of the lowest. Figs. 8(e,f) show two renderings of the radiance map using a new tone reproduction algorithm [23]. Although the rightmost stained glass window has radiance values over a thousand times higher than the darker areas in the rafters, these renderings exhibit detail in both areas.

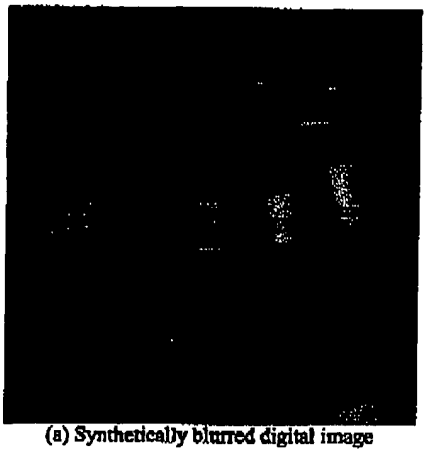
Figure 9 demonstrates two applications of the techniques presented in this paper: accurate signal processing and virtual photography. The task is to simulate the effects of motion blur caused by moving the camera during the exposure. Fig. 9(a) shows the results of convolving an actual, low-dynamic range photograph with a 37×1 pixel box filter to simulate horizontal motion blur. Fig. 9(b) shows the results of applying this same filter to the high dynamic range radiance map, and then sending this filtered radiance map back through the recovered film response functions using the same exposure time Δt as in the actual photograph. Because we are seeing this image through the actual image response curves, the two left images are tonally consistent with each other. However, there is a large difference between these two images near the bright spots. In the photograph, the bright radiance values have been clamped to the maximum pixel values by the response function. As a result, these clamped values blur with lower neighboring values and fail to saturate the image in the final result, giving a muddy appearance.

In Fig. 9(b), the extremely high pixel values were represented properly in the radiance map and thus remained at values above the level of the response function's saturation point within most of the blurred region. As a result, the resulting virtual photograph exhibits several crisply-defined saturated regions.

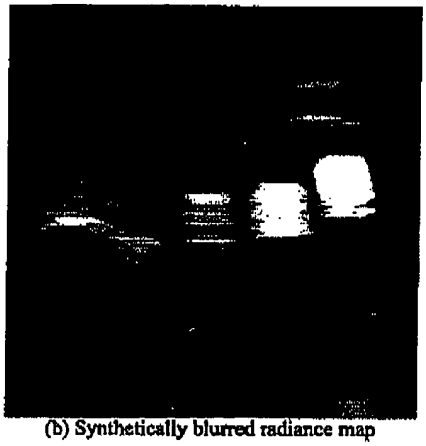
Fig. 9(c) is an actual photograph with real motion blur induced by spinning the camera on the tripod during the exposure, which is equal in duration to Fig. 9(a) and the exposure simulated in Fig. 9(b). Clearly, in the bright regions, the blurring effect is qualitatively similar to the synthetic blur in 9(b) but not 9(a). The precise shape of the real motion blur is curved and was not modeled for this demonstration.

[10] This feature of the PhotoCD process is called "Scene Balance Adjustment", or SBA.

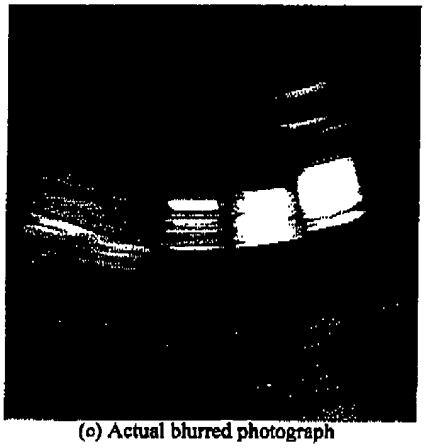
[11] This is far less of a problem for cinematic applications, in which the film sprocket holes are used to expose and scan precisely the same area of each frame.



(a) Synthetically blurred digital image



(b) Synthetically blurred radiance map



(c) Actual blurred photograph

Figure 9: (a) *Synthetic motion blur applied to one of the original digitized photographs. The bright values in the windows are clamped before the processing, producing mostly unsaturated values in the blurred regions.* (b) *Synthetic motion blur applied to a recovered high-dynamic range radiance map, then virtually re-photographed through the recovered film response curves. The radiance values are clamped to the display device* after *the processing, allowing pixels to remain saturated in the window regions.* (c) *Real motion blur created by rotating the camera on the tripod during the exposure, which is much more consistent with (b) than (a).*

4 Conclusion

We have presented a simple, practical, robust and accurate method of recovering high dynamic range radiance maps from ordinary photographs. Our method uses the constraint of sensor reciprocity to derive the response function and relative radiance values directly from a set of images taken with different exposures. This work has a wide variety of applications in the areas of image-based modeling and rendering, image processing, and image compositing, a few of which we have demonstrated. It is our hope that this work will be able to help both researchers and practitioners of computer graphics make much more effective use of digitized photographs.

Acknowledgments

The authors wish to thank Tim Hawkins, Carlo Séquin, David Forsyth, Steve Chenney, Chris Healey, and our reviewers for their valuable help in revising this paper. This research was supported by a Multidisciplinary University Research Initiative on three dimensional direct visualization from ONR and BMDO, grant FDN00014-96-1-1200.

References

[1] ADAMS, A. *Basic Photo*, 1st ed. Morgan & Morgan, Hastings-on-Hudson, New York, 1970.

[2] CHEN, E. QuickTime VR - an image-based approach to virtual environment navigation. In *SIGGRAPH '95* (1995).

[3] DEBEVEC, P. E., TAYLOR, C. J., AND MALIK, J. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *SIGGRAPH '96* (August 1996), pp. 11–20.

[4] FAUGERAS, O. *Three-Dimensional Computer Vision*. MIT Press, 1993.

[5] FERWERDA, J. A., PATTANAIK, S. N., SHIRLEY, P., AND GREENBERG, D. P. A model of visual adaptation for realistic image synthesis. In *SIGGRAPH '96* (1996), pp. 249–258.

[6] GORTLER, S. J., GRZESZCZUK, R., SZELISKI, R., AND COHEN, M. F. The Lumigraph. In *SIGGRAPH '96* (1996), pp. 43–54.

[7] HORN, B. K. P. *Robot Vision*. MIT Press, Cambridge, Mass., 1986, ch. 10, pp. 206–208.

[8] JAMES, T., Ed. *The Theory of the Photographic Process*. Macmillan, New York, 1977.

[9] KAUFMAN, J. E., Ed. *IES Lighting Handbook; the standard lighting guide*, 7th ed. Illuminating Engineering Society, New York, 1987, p. 24.

[10] KOLB, C., MITCHELL, D., AND HANRAHAN, P. A realistic camera model for computer graphics. In *SIGGRAPH '95* (1995).

[11] LAVEAU, S., AND FAUGERAS, O. 3-D scene representation as a collection of images. In *Proceedings of 12th International Conference on Pattern Recognition* (1994), vol. 1, pp. 689–691.

[12] LEVOY, M., AND HANRAHAN, P. Light field rendering. In *SIGGRAPH '96* (1996), pp. 31–42.

[13] MADDEN, B. C. Extended intensity range imaging. Tech. rep., GRASP Laboratory, University of Pennsylvania, 1993.

[14] MANN, S., AND PICARD, R. W. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In *Proceedings of IS&T 46th annual conference* (May 1995), pp. 422–428.

[15] MCMILLAN, L., AND BISHOP, G. Plenoptic Modeling: An image-based rendering system. In *SIGGRAPH '95* (1995).

[16] SCHLICK, C. Quantization techniques for visualization of high dynamic range pictures. In *Fifth Eurographics Workshop on Rendering (Darmstadt, Germany)* (June 1994), pp. 7–18.

[17] SZELISKI, R. Image mosaicing for tele-reality applications. In *IEEE Computer Graphics and Applications* (1996).

[18] TANI, T. *Photographic sensitivity : theory and mechanisms*. Oxford University Press, New York, 1995.

[19] THEUWISSEN, A. J. P. *Solid-state imaging with charge-coupled devices*. Kluwer Academic Publishers, Dordrecht; Boston, 1995.

[20] TUMBLIN, J., AND RUSHMEIER, H. Tone reproduction for realistic images. *IEEE Computer Graphics and Applications 13*, 6 (1993), 42–48.

[21] WARD, G. J. Measuring and modeling anisotropic reflection. In *SIGGRAPH '92* (July 1992), pp. 265–272.

[22] WARD, G. J. The radiance lighting simulation and rendering system. In *SIGGRAPH '94* (July 1994), pp. 459–472.

[23] WARD, G. J., RUSHMEIER, H., AND PIATKO, C. A visibility matching tone reproduction operator for high dynamic range scenes. Tech. Rep. LBNL-39882, Lawrence Berkeley National Laboratory, March 1997.

A Matlab Code

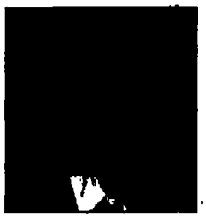
Here is the MATLAB code used to solve the linear system that minimizes the objective function $\mathcal{O}$ in Equation 3. Given a set of observed pixel values in a set of images with known exposures, this routine reconstructs the imaging response curve and the radiance values for the given pixels. The weighting function $w(z)$ is found in Equation 4.

```
%
% gsolve.m - Solve for imaging system response function
%
% Given a set of pixel values observed for several pixels in several
% images with different exposure times, this function returns the
% imaging system's response function g as well as the log film irradiance
% values for the observed pixels.
%
% Assumes:
%
%   Zmin = 0
%   Zmax = 255
%
% Arguments:
%
%   Z(i,j) is the pixel values of pixel location number i in image j
%   B(j)   is the log delta t, or log shutter speed, for image j
%   l      is lambda, the constant that determines the amount of smoothness
%   w(z)   is the weighting function value for pixel value z
%
% Returns:
%
%   g(z)   is the log exposure corresponding to pixel value z
%   lE(i)  is the log film irradiance at pixel location i
%

function [g,lE]=gsolve(Z,B,l,w)

n = 256;

A = sparse(size(Z,1)*size(Z,2)+n+1,n+size(Z,1));
b = sparse(size(A,1),1);

% Include the data-fitting equations

k = 1;
for i=1:size(Z,1)
  for j=1:size(Z,2)
    wij = w(Z(i,j)+1);
    A(k,Z(i,j)+1) = wij;  A(k,n+i) = -wij;       b(k,1) = wij * B(j,1);
    k=k+1;
  end
end

% Fix the curve by setting its middle value to 0

A(k,129) = 1;
k=k+1;

% Include the smoothness equations

for i=1:n-2
  A(k,i)=l*w(i+1);          A(k,i+1)=-2*l*w(i+1);  A(k,i+2)=l*w(i+1);
  k=k+1;
end

% Solve the system using SVD

x = A\b;

g = x(1:n);
lE = x(n+1:size(x,1));
```

JIM BLINN'S CORNER

Compositing, Part I: Theory

James F. Blinn, *California Institute of Technology*



Associating a pixel's color with its opacity is the basis for a compositing function that is simple, elegant, and general. But there are more reasons than mere prettiness to store pixels this way.

My currently favorite journalistic quote comes from a magazine called *Morph's Outpost on the Digital Frontier*. They refer to the operation of avoiding jaggies as "anti-aliening." Either this was a typo or they thought of the jaggies as aliens. This got me thinking about ways to get rid of these creatures —the offspring of 3D geometry and raster displays.

One of the most important anti-aliasing tools in computer graphics comes from a generalization of the simple act of storing a pixel into a frame buffer. Several people simultaneously discovered the usefulness of this operation, so it goes by several names: matting, image compositing, alpha blending, overlaying, or lerping. It was most completely codified in a paper by Porter and Duff,[1] where they call it the "over" operator. In this column I'm going to show a new way to derive Porter and Duff's "over" operator and describe some implementation details that I've found useful. In a later column I'll go into some of the subtleties of how this operator works with integer pixel arithmetic.

The basic idea

The simplest form of compositing goes as follows. Say we want to overlay a foreground image on some background image. The foreground image only covers a part of the background; pixels inside the foreground shape will completely replace the corresponding background pixels, and pixels outside the shape leave the background pixels intact.

If we want anti-aliened edges, though, things are a bit more complicated. Pixels on the edge of the shape only partially cover the background pixels. If the shape is to be properly anti-aliened, we must blend the foreground color, F, and background color, B, according to the fraction α. This value represents the percentage of the pixel covered by color F. The standard way to calculate this is to find the geometric area covered by F. This implements a simple box filter for anti-aliening. More accurate filters can be used, but I'll stick to the box for now.

Now let's get down to algebra. F and B are each three-element vectors representing the red, green, and blue components of a pixel. Ordinary vector algebra applies. The new color in the frame buffer is

$$B_{new} = (1 - \alpha)\, B_{old} + \alpha F$$

which can be more efficiently calculated as

$$B_{new} = B_{old} + \alpha(F - B_{old})$$

You can actually use the value of α for a variety of things. In addition to its anti-aliening function, it can represent transparent objects or establish a global fade amount. For this reason, the α value also goes by various names: coverage amount, opacity, or simply alpha. You can also think of it as 1 minus the transparency of the pixel. I'm going to call it opacity for now. If it's 0, the new pixel is transparent and does not affect the frame buffer. If it's 1, the new pixel is opaque and completely replaces the current frame buffer color.

Next, suppose that we want to layer another object on top of our image. We just blend in the new object's color, which I'll call G, on top of our current background image using its opacity β,

$$B_{newnew} = (1 - \beta)B_{new} + \beta G$$

We can keep on plastering stuff on top of our image until we are happy. This is the essence of 2-1/2D rendering, also known as the painter's algorithm or temporal priority.

For most rendering purposes I've been able to provide this as the only necessary accessing operation into the frame buffer. But it's not quite general enough.

Associativity

There is another intriguing generalization here. Both F and G have an opacity, but B doesn't. Does it even mean anything to composite into a pixel that already has an opacity? Yes. Consider the following scenario. Suppose we have the images F and G, but haven't yet decided what to use for a background. Let's see if we can merge F and G into one image, H, that we can store away and later overlay on B to get the same result. If we denote the compositing operation with the symbol &, what we want is

$$(B \,\&\, F) \,\&\, G = B \,\&\, (F \,\&\, G)$$

In other words we want to make compositing *associative*.

How can we define H = F & G to make this work out? We want to calculate a new pixel color H and opacity γ in terms of colors F and G and their own opacities α and β. Plug in the definitions:

0272-17-1694/94.00 © 1994 IEEE

$$(1-\beta)\left((1-\alpha)B+\alpha F\right)+\beta G=(1-\gamma)B+\gamma H$$

Rearrange the left side to get

$$(1-\alpha)(1-\beta)B+\left(\alpha(1-\beta)F+\beta G\right)=(1-\gamma)B+\gamma H$$

Since we want this to work for arbitrary backgrounds B, we can split this into two equations by equating the B coefficients and the non-B coefficients:

$$(1-\alpha)(1-\beta)=1-\gamma$$
$$\alpha(1-\beta)F+\beta G=\gamma H$$

The first of these gives us

$$\gamma=\alpha+\beta-\alpha\beta$$

The second equation gives us

$$H=\left(\alpha(1-\beta)F+\beta G\right)/\gamma$$

With a little fiddling this turns into

$$H=(1-\beta/\gamma)F+(\beta/\gamma)G$$

This gives us a definition for how to composite two colors, each of which has its own opacity.

Let's play with this a bit. If we composite G over a totally opaque color F, what is the result? Plug $\alpha=1$ into the above and we get

$$\gamma=1$$
$$H=(1-\beta)F+\beta G$$

In other words, our more general compositing operation boils down to the basic one if we assume our background has its own opacity value, which happens to be 1.

Now let's try overlaying a completely opaque color G on F. Plug in $\beta=1$ with an arbitrary α and we discover

$$\gamma=1$$
$$H=G$$

independent of α, as we expect.

Another form of association

The above definition of H is a bit complicated. Fortunately, there is a better way. One of the key insights in the Porter and Duff paper is that F shows up in the compositing formula only when multiplied by α, and G appears only when multiplied by β. Why not simply represent the pixel with the colors already premultiplied by their opacity? This representation is usually referred to as having the opacity *associated* with the color. I'll write (for the time being) an associated pixel color with a tilde over it. We have

$$\tilde{F}=\alpha F$$
$$\tilde{G}=\beta G$$
$$\tilde{H}=\gamma H$$

An associated color is just a regular color composited onto black—that is, if you displayed it directly by itself, you would get the correct anti-aliened image. (Is the joke worn out now? OK, I'll use the real word again.) Note that if the opacity equals 1, an associated color is the same as an unassociated color.

Using these definitions in the general compositing function and doing a bit of algebraic fiddling we get

$$\tilde{H}=(1-\beta)\tilde{F}+\tilde{G}$$
$$\gamma=(1-\beta)\alpha+\beta$$

This is a bit less arithmetic than our earlier definition, but what makes it particularly pretty is that we are now doing exactly the same arithmetic on the opacity components of a pixel as we are doing on the (associated) color components. This is simple, elegant, and general.

More reasons to associate

There are more reasons to store associated pixel colors than mere prettiness of the compositing formula. For one thing, some intensity calculation algorithms directly generate associated pixel colors. Additionally, we must use associated colors for any filtering or interpolation operations. Let's see why.

Antialiasing by subsampling

One typical way to do antialiasing is by subsampling. You calculate an image using point sampling at, say, four times your final resolution in x and y, and then downsample to get your final result. There are still aliases, but you have pushed them up into higher frequencies.

How does this work with our scheme here? You can consider each final pixel as broken into a 4×4 grid of subpixel cells, each containing a color and an opacity flag. Initialize these all to 0. Then, whenever your renderer writes a color to a subpixel cell, have it also set the opacity flag to 1. After rendering, sum up the 16 opacity flags within the pixel and call the result N. The net opacity for the pixel is $N/16$. Next, sum up the 16 color cells in the pixel. The average color of the pixel is this sum divided by N, the number of cells colored. But the *associated* color is even more simply calculated as the color sum divided by 16, $(\text{sum}/N) * (N/16) = \text{sum}/16$. You can then composite this associated color using the calculated opacity, $N/16$. In other words, the net associated pixel color and opacity is the sum of the subpixels divided by 16.

This works even better if your renderer is scan-line oriented—that is, it visits each pixel once in order left to right, top to bottom. You don't need individual subpixel cells. Just accumulate the color and opacity into a single pixel cell and divide by 16. In practice, I implement this with a scan-line buffer of pixel cells of length equal to the output picture. During scan-line processing, each high-resolution pixel gener-

ated simply adds its value to cell number $x/4$. Then, every four scan lines, I purge this buffer by dividing its contents by 16 and compositing it with the background using the associated compositing formula. Then I zero the buffer in preparation for the next four scan lines.

Clouds

The cloud simulation I used for Saturn's rings[1] generates a pixel's brightness as a product of the color of a cloud particle times the probability of a particle being both present in the pixel and illuminated. We can now recognize this as an associated color. The Saturn cloud simulation also generates a transparency value based on probabilities of blocking particles. The compositing operator I described for the simulation[2] is just the associated composition operator, but I didn't recognize it as such at first. Originally, I actually divided the color by the opacity before passing it to an unassociated compositing routine. Live and learn.

Filtering

Suppose we want to filter an image that has opacities at each pixel. Do we filter the unassociated colors F (this was my first thought), or do we filter the associated colors $\hat{F}$? To find out, consider the following thought experiment.

Let's downsample a scan line by a factor of two in the x direction by simply averaging successive pairs of pixels. Then let's overlay the result on an opaque background B. We want to arrange things so that downsampling and overlaying generate the same color as overlaying and downsampling. Let's follow the adventures of a typical pixel pair F (with opacity α) and G (with opacity β). Note that F and G are side by side here, not on top of each other as in our earlier examples.

First, try overlaying and then downsampling. Overlay (F, α) on B, getting $\alpha F + (1 - \alpha)B$. Overlay (G, β) on B, getting $\alpha G + (1 - \beta)B$. These two pixels are now opaque. Now downsample by averaging these results. The color will be

$$\frac{\alpha}{2}F + \frac{\beta}{2}G + \frac{2-\alpha-\beta}{2}B$$

As long as you composite first, it actually doesn't matter if you do it associated or unassociated.

Next, let's do this in the other order: downsampling first, then overlaying. Downsampling the unassociated colors and opacity, we get

$$\text{color} = (F + G)/2;\ \text{opacity} = (\alpha + \beta)/2$$

Now overlay this on B using the unassociated color compositing function to get

$$\left(1 - \frac{\alpha+\beta}{2}\right)B + \left(\frac{\alpha+\beta}{2}\right)\frac{F+G}{2} = \frac{\alpha+\beta}{4}F + \frac{\alpha+\beta}{4}G + \frac{2-\alpha-\beta}{2}B$$

—the wrong answer.

Now let's do this with associated colors. Downsample αF and βG:

$$\text{color} = (\alpha F + \beta G)/2;\ \text{opacity} = (\alpha + \beta)/2$$

Now overlay this on B using the associated compositing function to get

$$\left(1 - \frac{\alpha+\beta}{2}\right)B + \frac{\alpha F + \beta G}{2} = \frac{\alpha}{2}F + \frac{\beta}{2}G + \frac{2-\alpha-\beta}{2}B$$

—the right answer.

To reiterate, downsampling and, in fact, *all* filtering operations should operate on arrays of associated pixel colors as well as, of course, on the array of opacity values.

Interpolation

Here's another example. Suppose we are doing Gouraud interpolation across a polygon. Each vertex has a color, and we do the standard interpolation of vertex colors to get the colors inside the polygon. Now, what if the vertices have opacities as well? We simply interpolate them in a similar manner. But should we interpolate unassociated colors or associated colors? (I'll bet you can guess.)

Actually, this might seem a little open to interpretation. After all, Gouraud interpolation is itself an approximation of a more accurate curved-surface-shading function. Who's to say what the correct interpolation amount is? Well, consider the following: Interpolation is another form of filtering. Suppose we wanted to expand an image two times by interpolating between each pixel pair. We would again like this to look the same if we interpolated and then overlayed on a background or if we overlayed first and interpolated second.

Going back to polygons, we might have a scan line with the colors (F, α) on one end and (G, β) on the other. We want the inside colors to look the same when overlayed onto a background. We want to interpolate and then overlay over B, and we want to make this the same as overlaying and then interpolating.

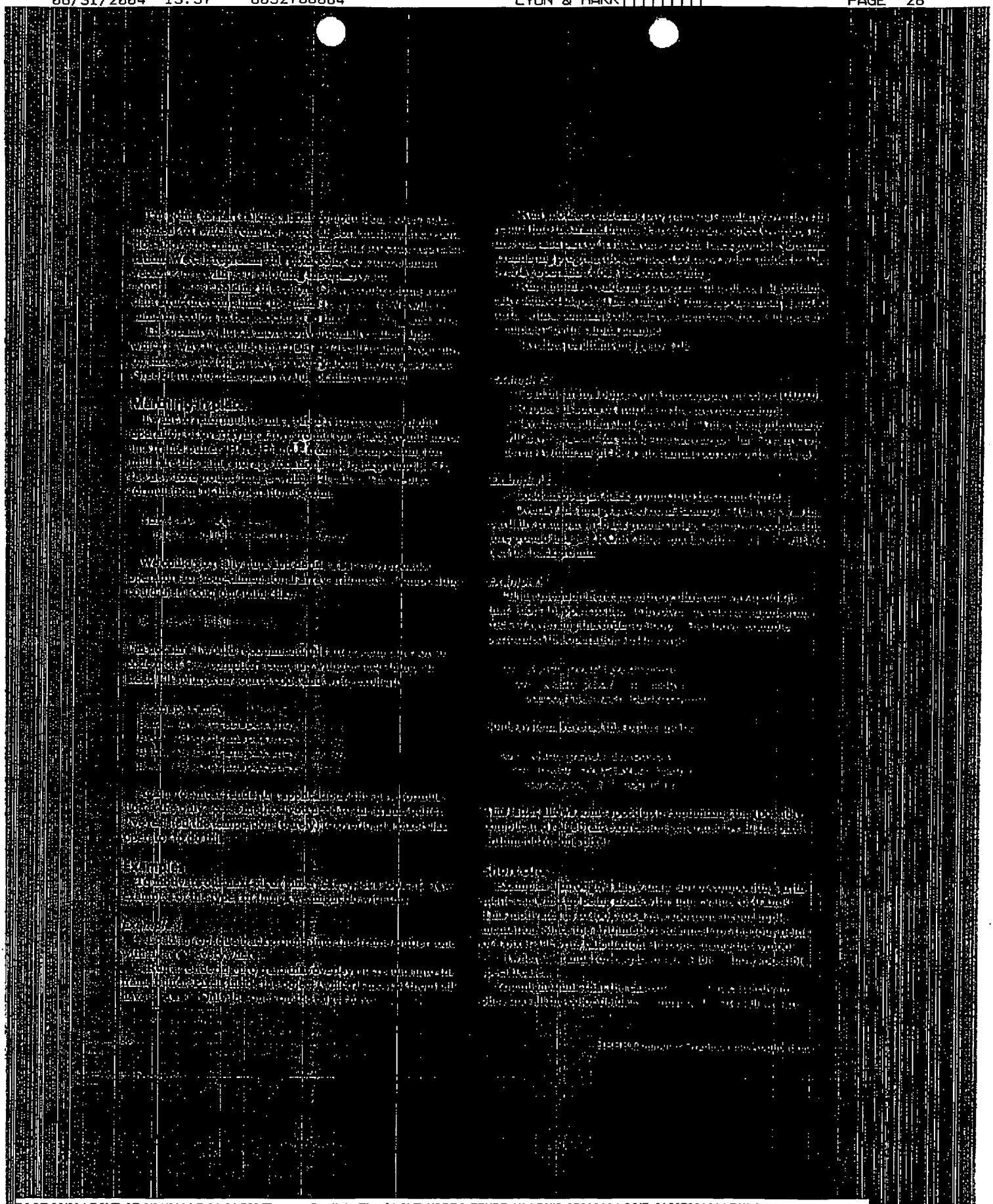
You can do the algebra yourself. Does it look familiar? It's just the same as the filtering example, leading us to the conclusion that Gouraud interpolation should also be done on associated pixel colors.

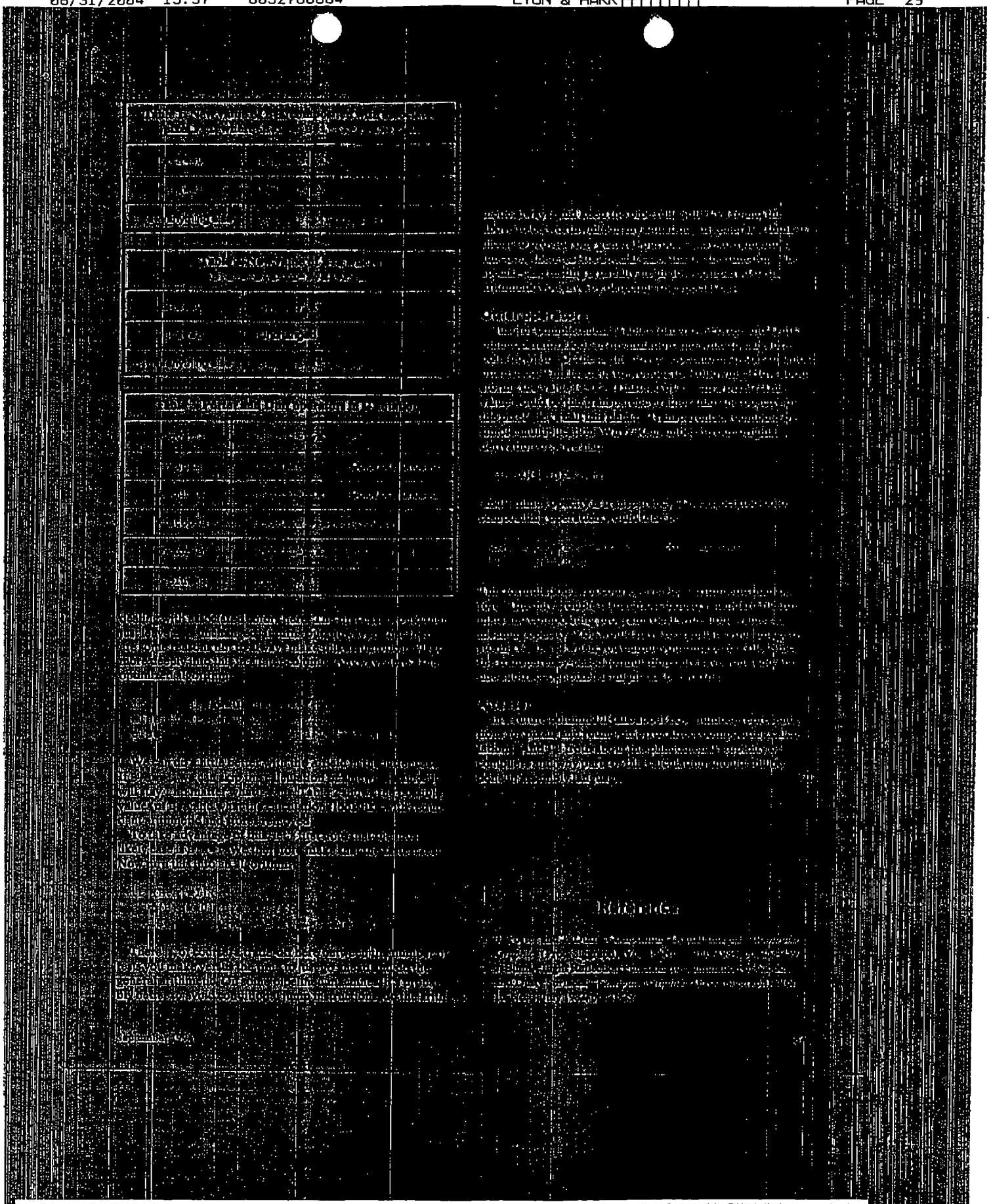
Computer notation

Each pixel has a red, green, and blue color and an opacity α. Since we like associated colors so much, we will represent a pixel by the quadruple:

$$(\alpha F_{red}, \alpha F_{green}, \alpha F_{blue}, \alpha)$$

This looks suspiciously like homogeneous coordinates. I've tried real hard, but for the life of me I can't figure out any use for this observation.





This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☒ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.